Amendments to the Claims

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims

1. (Currently Amended) A software diagnostics platform comprising:

   a command interface adapted to receiving commands and outputting results;

   an engine adapted to running a test sequence;

   an output driver adapted to timestamping an outgoing message and storing said outgoing message;

   an input driver adapted to timestamping an incoming message and storing said incoming message; and

   an analysis routine adapted to analyzing said outgoing message and said incoming message to determine at least one performance parameter;

   said software diagnostics platform being operable on an embedded processor being operable on a single board ~~having a fewer number of input ports than said embedded processor~~.

2. (Previously Presented) The software diagnostics platform of claim 1 wherein said command interface is operable to communicate via a terminal interface.

3. (Previously Presented) The software diagnostics platform of claim 1 further comprising:

   a host program operable on a host system, said host program having a graphical user interface.

4. (Previously Presented) The software diagnostics platform of claim 1 wherein said test sequence comprises a single test routine.

5. (Previously Presented) The software diagnostics platform of claim 1 wherein said test sequence comprises a test routine that is repeated for a predetermined period of time.

6. (Previously Presented) The software diagnostics platform of claim 1 wherein said test sequence comprises multiple threads of commands.

7. (Currently Amended) The software diagnostics platform of claim ~~1 wherein input driver is further adapted to validate said incoming message~~ 6, at least two of said multiple threads of commands being configured to operate on a single port in parallel.

8. (Currently Amended) The software diagnostics platform of claim 1 wherein said ~~analysis~~ performance parameter comprises determining one of a group composed of: message transfer time, average message transfer time, and average data throughput per unit time.
9. (Previously Presented) The software diagnostics platform of claim 1 further comprising an initiator adapted to determine if an I/O device is present.
10. (Previously Presented) The software diagnostics platform of claim 9 wherein said initiator is further adapted to perform a diagnostic routine with said I/O device.
11. (Currently Amended) A system comprising:

a device with an embedded processor, said device having a specific function~~, said embedded processor having a first number of output ports and said device having fewer output ports than said embedded processor~~;

a first software system operable to run on said embedded processor and enable said device to perform said specific function; and

a second software system operable to run on said embedded processor, said second software system comprising:

a command interface adapted to receiving commands and outputting results;

an engine adapted to running a test sequence, said test sequence comprising at least two threads capable of operating on a port in parallel;

an output driver adapted to timestamping an outgoing message and storing said outgoing message, said output driver being configured to send said outgoing message to each of said first number of output ports;

an input driver adapted to timestamping an incoming message and storing said incoming message; and

an analysis routine adapted to analyzing said outgoing message and said incoming message.

12. (Previously Presented) The system of claim 11 wherein said command interface is operable to communicate via a terminal interface.
13. (Previously Presented) The system of claim 11 further comprising:

a host program operable on a host system, said host program having a graphical user interface.

14. (Previously Presented) The system of claim 11 wherein said test sequence comprises a single test routine.

15. (Previously Presented) The system of claim 11 wherein said test sequence comprises a test routine that is repeated for a predetermined period of time.
16. (Previously Presented) The system of claim 11 wherein said test sequence comprises multiple threads of commands.
17. (Previously Presented) The system of claim 11 wherein input driver is further adapted to validate said incoming message.
18. (Previously Presented) The system of claim 11 wherein said analysis comprises determining one of a group composed of: message transfer time, average message transfer time, and average data throughput per unit time.
19. (Previously Presented) The system of claim 11 further comprising an initiator adapted to determine if an I/O device is present.
20. (Previously Presented) The system of claim 19 wherein said initiator is further adapted to perform a diagnostic routine with said I/O device.
21. (Currently Amended) A test system comprising:

a reusable test sequence;

a first command interpreter adapted to interpret said reusable test sequence, said first command interpreter being adapted to operate on a first embedded processor, said first embedded processor being in a first circuit having a first functionality; and

a second command interpreter adapted to interpret said reusable test sequence, said second command interpreter being adapted to operate on a second embedded processor, said second embedded processor being in a second circuit having a second functionality, said second functionality being different from said first functionality;

wherein said first command interpreter and said second command interpreter each comprise:

a command interface adapted to receiving commands and outputting results;

an engine adapted to running a test sequence, said test sequence comprising at least two threads configured to operate in parallel on a single port;

an output driver adapted to timestamping an outgoing message and storing said outgoing message;

an input driver adapted to timestamping an incoming message and storing said incoming message; and

an analysis routine adapted to analyzing said outgoing message and said incoming message.

22. (Currently Amended) A method of developing a circuit having an embedded processor comprising:

designing a circuit having said embedded processor, said circuit having a predefined function;

assembling said circuit;

designing software operable on said embedded processor, said software adapted to enable said circuit to perform said predefined function;

loading said embedded processor with a test platform software comprising:

a command interface adapted to receiving commands and outputting results;

an engine adapted to running a test sequence, said test sequence comprising at least two threads configured to operate in parallel on a single port;

an output driver adapted to timestamping an outgoing message and storing said outgoing message;

an input driver adapted to timestamping an incoming message and storing said incoming message;

an analysis routine adapted to analyzing said outgoing message and said incoming message and create results; and

a display routine for displaying said results;

creating said test sequence;

transmitting said test sequence to said embedded processor loaded with said test platform software;

operating said test sequence on said embedded processor; and

analyzing said results.

23. (Currently Amended) A turn on software suite comprising:

a bootstrap code configured to start said software suite on an embedded processor, said embedded processor being embodied in a single board device having application specific board level circuitry, said embedded processor having a first number of input paths and a second number of output paths;

a test engine configured to create a plurality of test workers;

each of said test workers being configured to exercise one of said input path or output path, at least two of said test workers being configured to operate on one of said input paths in parallel;

said turn on software suite being capable of executing on said embedded processor and operating said plurality of test workers~~, said application specific board level circuitry comprising fewer input paths and fewer output paths than said embedded processor~~.